

毕业论文

**题 目 基于web的网上书城系统**

**学 院 电子信息与电气工程学院**

**专 业 电子信息工程**

**班 级 15级电信一班**

**姓 名 杨智妍**

**学 号 20151060111**

**指导教师 赵玉祥**

**提交日期 2018年5月20日**

**原创性声明**

**本人郑重声明：本人所呈交的论文是在指导教师的指导下独立进行研究所取得的成果。学位论文中凡是引用他人已经发表或未经发表的成果、数据、观点等均已明确注明出处。除文中已经注明引用的内容外，不包含任何其他个人或集体已经发表或撰写过的科研成果。**

**本声明的法律责任由本人承担。**

**论文作者签名： 年 月 日**

**论文指导教师签名：**

# 摘要

随着前端各种新兴技术的崛起，我们编写前端已经不仅仅局限于html，css，js，而是有了更多的选择，用更加简洁的代码就可以实现更加完美的功能。“基于web的网上书城系统”的前端开发就是采用vue和基于vue开发的桌面组件库element ui，后台采用基于nodejs的express框架，数据库采用mongodb。

“网上书城”实现的功能是书城的前端界面设计，前端交互界面基于vue开发，书城主要完成了三个系统模块的实现，1.实现用户读书心得交流，2.实现电子书排行查看和电子书下载，3.图书商城中加入购物车和付款。在开发过程中，主要需要注意的可分为四个方向，一是对加载速度的要求，不管从用户还是开发者的角度，加载速度都应该被重视；二是对界面的排版，首先界面不能让用户感到反感，因此要适应大众的审美观；三是图片的选择，图片不仅影响美观，而且如果图片太大还会影响加载速度；四就是留意配色，配色的好坏影响着界面的美观与否。

后台管理系统是基于element ui框架开发的，实现了后端商品管理界面设计功能。element ui 的优点在于可以方便的开发出功能丰富而且美观的后台管理界面，不需要写很多的CSS代码就可以实现布局，因为框架的开发就在于对代码的封装，而使用框架开发就只需要调用就好。

**关键词 :** 书城 Vue element ui nodejs mongodb express

# ABSTRACT

With the rise of various emerging front-end technologies, we are no longer limited to writing front-end HTML, CSS and js, but have more choices, and can achieve more perfect functions with more concise code. The front-end development of "web-based online book city system" is the desktop component library element UI developed based on vue and vue, the back-end is the express framework based on nodejs, and the database is mongodb.

The function of "online book city" is the front-end interface design of book city. The front-end interactive interface is developed based on vue. The book city mainly completes the realization of three system modules:1. To achieve the user reading experience exchange, 2. To achieve the e-book ranking view and e-book download, 3. Add shopping cart and payment to book mall. In the development process, the main need to pay attention to can be divided into four directions, firstly, the load speed requirements, whether from the perspective of users or developers, load speed should be paid attention to.Secondly, the layout of the interface, first of all, the interface can not make the user feel disgusted, so to adapt to the public aesthetic. Thirdly, the choice of pictures, pictures not only affect the appearance, and if the picture is too big will affect the loading speed; Forthly, pay attention to color matching, color matching affects the interface is beautiful or not.

Background management system is based on element UI framework development, the realization of the back-end commodity management interface design function. The advantage of element UI is that it can easily develop a rich and beautiful background management interface without writing a lot of CSS code to achieve the layout. Because the development of the framework is to encapsulate the code, while the development of the framework only needs to be called.

**Key words :** Book Mall Vue Element UI Nodejs Mongodb Express

目录

[摘要 I](#_Toc7357828)

[ABSTRACT II](#_Toc7357829)

[第1章 绪论 - 1 -](#_Toc7357830)

[1.1 研究背景 - 1 -](#_Toc7357831)

[1.2 研究目的及意义 - 1 -](#_Toc7357832)

[1.3 开发工具 - 2 -](#_Toc7357833)

[第2章 技术选型和设计方案 - 3 -](#_Toc7357834)

[2.1 涉及的开发技术 - 3 -](#_Toc7357835)

[2.2 书城功能结构图 - 6 -](#_Toc7357836)

[第3章 书城前端功能实现与展示 - 8 -](#_Toc7357837)

[3.1 登录注册功能 - 8 -](#_Toc7357838)

[3.2 首页面功能 - 9 -](#_Toc7357839)

[3.3 好书分享功能 - 10 -](#_Toc7357840)

[3.4 电子书页面功能 - 11 -](#_Toc7357841)

[3.5 书城页面功能 - 12 -](#_Toc7357842)

[3.6 购物系统功能分析 - 13 -](#_Toc7357843)

[第4章 后端书籍管理功能与实现 - 18 -](#_Toc7357844)

[4.1 后端书籍管理界面原理分析 - 18 -](#_Toc7357845)

[4.2 后端书籍管理界面分析 - 18 -](#_Toc7357846)

[第5章 项目疑难点和总结 - 20 -](#_Toc7357847)

[5.1 书城前端界面系统问题分析及解决 - 21 -](#_Toc7357848)

[5.2 后端商品管理界面问题分析及解决 - 23 -](#_Toc7357849)

[5.3 总结 - 24 -](#_Toc7357850)

[参考文献 - 25 -](#_Toc7357851)

[致 谢 - 26 -](#_Toc7357852)

# 

# 绪论

## 1.1 研究背景

早期的网页开发是由后端主导的，前端能做的也就是操作一下DOM。2006年 John Resig发布了jQuery，用于操作DOM，其优雅的语法、符合直觉的事件驱动型的编程思维使其极易上手，因此很快风靡全球，大量基于jQuery的插件构成了一个庞大的生态系统。起初制约Web开发从后到前的因素很简单，就是前端很多事干不了或干不好，当时的浏览器性能弱，标准化程度低。特别是占据大量市场份额的IE，不仅ugly，并且buggy。于2008年问世的谷歌V8引擎改变了这一局面。现代浏览器的崛起终结了微软的垄断时代，前端的计算能力一下子变得过剩了。标准组织也非常配合的在2009年发布了第五代JavaScript，前端的装备得到了整体性的提高，前端界如同改革开放走进了一个令人目不暇接的新时代。2009年AngularJS诞生，随后被谷歌收购。2010年backbone.js诞生。2011年React和Ember诞生。2014年Vue.js诞生……前后端分离可谓大势所趋。后端只负责数据，前端负责其余工作，这种分工模式一定是更清晰也更高效的。随着基础设置的不断完善以及代码封装层级的不断提高，使得前端一个人能够完成的事越来越多，这是技术积累的必然结果。

2009年Ryan Dahl发布了node，node是一个基于V8引擎的服务端JavaScript运行环境，类似于一个虚拟机，也就是说js在服务端语言中有了一席之地。如果说ajax是前端的第一次飞跃，那么node可算作前端的第二次飞跃。它意味着JavaScript走出了浏览器的藩篱，迈出了全端化的第一步，一种流行的组合就是利用Express和Mongo搭建服务端程序。“网上书城系统”前端采用vue+element ui进行开发，后端采用express+node进行搭建，符合现代技术发展方向，且为爱看书的书友提供了更多的便利。

## 1.2 研究目的及意义

“网上书城系统”在现如今网络技术非常发达的情况下具有很大的开发意义，从阅读和购书的方便简洁带给人们更为广泛的方便性。所以根据人们的所需，开发“网上书城系统”是一个很好的研究方向。

本次毕业设计的目的是建立一个动态的、交互的在线交流读书心得和购书的网站。用户可以通过网站来分享自己的读书心得，也可以来购书。通俗地讲，我们可以把传统的商店直接“搬”回家，利用Internet直接购买自己需要的商品或者享受自己需要的服务。专业地讲，它是交易双方从洽谈、签约以及贷款的支付、交货通知等整个交易过程通过Internet、web和购物界面的技术化的B2C模式。实现电子商务的信息流、资金流和物流三方面的内容。

## 1.3 开发工具

（1）VS Code

前端开发工具很多，有最基本开发工具Notepad++，适用于入门开发者，高级开发工具Webstorm，方便于前后端一起开发的MyEclipse工具等，Visual Studio Code（简称VS Code）是一个由微软开发的，同时支持Windows、Linux、和macOS系统且开放源代码的代码编辑器，它支持测试，并内置了Git 版本控制功能，同时也具有开发环境功能，例如代码补全（类似于 IntelliSense）、代码片段、和代码重构等，该编辑器支持用户个性化配置，例如改变主题颜色、键盘快捷方式等各种属性和参数，还在编辑器中内置了扩展程序管理的功能。

（2）Firefox、Google 、IE浏览器

前端开发最重要的一部分也就是测试，测试离不开的就是各大主流浏览器，由于不同浏览器的内核是不同的，所以有时候出现的bug也是不同的，因此通过各大浏览器的测试，才能完美的解决兼容性问题。

（3）PhotoShop

PhotoShop，简称PS，是作图，修图必不可少的工具，尤其是在现在前端开发对于图片色彩等的搭配，选择自己想要的色彩，抠图、P图等操作都需要PS来完成。

（4）开发者模式

开发者模式是作为开发者不可缺少的调试代码的平台，浏览器右击选择选择检查或者键盘按F12可以进入浏览器的开发者模式，当写网页动态页面的时候总会需要调试，平台会进行错误提示，或者样式不对时，可以点击检查元素，点击想要查看的元素将定位到元素的位置，给开发者带来很大的方便。

# 技术选型和设计方案

## 2.1 涉及的开发技术

本次设计从页面到细节，从静态到动态，然后到后台数据的交互，所涉及的技术有Vue，ECMAScript，LocalStorage/SessionStorage ，Vuex，Axios，Vue Router，Element UI， Node.js, Express, MongoDB, Webpack等。

2.1.1 Vue

vue是随着时代进步更新的一种方便开发者的前端框架。在学习成本方面，vue是用于构建用户界面的渐进式框架，与其它大型框架不同的是，vue被设计为可以自底向上的逐层应用。Vue的核心库只关注视图层，不仅易上手，还便于与第三方库或既有项目整合。另一方面，当与现代化的工具链以及各种支持类库结合使用时，vue也完全能够为复杂的单页面提供驱动。

2.1.2 ECMAScript

ECMAScript 6.0（以下简称 ES6）是 JavaScript 语言的下一代标准，在 2015 年 6 月正式发布了。使得 JavaScript 语言可以用来编写复杂的大型应用程序，成为企业级的开发语言。与之前的ES5相比，ES6新增了新特性，给开发者带来了更多的便利，例如变量定义中的块级作用域变量和只读变量，函数中给形参函数设置默认值，箭头函数，关于数组的展开运算，数组和对象的解构赋值，用来表独一无二值的原始数据类型Symbol，类似于数组，但是成员的值都是唯一的，没有重复的值的数据结构Set，类似于对象，也是键值对的集合，但是“键”的范围不限于字符串，各种类型的值都可以当做键的数据结构Map，用于修改某些操作的默认行为的Proxy等等。

2.1.3 LocalStorage/SessionStorage

html5 中的 web Storage 包括了两种存储方式：sessionStorage 和 localStorage。 sessionStorage 用于本地存储一个会话（session）中的数据，这些数据只有在同一个会话中的页面才能访问，会话结束后数据也随之销毁。localStorage 用于存储一个域名下的需要永久存在在本地的数据，这些数据可以被一直访问，直到这些数据被删除。因此sessionStorage 和 localStorage 的主要区别在于他们存储数据的生命周期，sessionStorage 存储的数据的生命周期是一个会话，而 localStorage 存储的数据的生命周期是永久，直到被主动删除，否则数据永远不会过期的。

2.1.4 Vuex

每个Vuex应用的核心就是store（仓库）。可以等同于一个容器，包含着应用中的大部分状态(state)。Vuex中的状态存储是响应式的，当我们的vue组件从store中读取状态的时候，如果store中的状态发生变化，那么响应的组件也会相应得得到高效更新；而且，我们不能直接改变store中的状态，改变store中的状态的唯一途径就是显式提交(commit)mutation。这样使得我们可以方便地去跟踪到每一个状态的变化，从而让我们能够实现一些工具帮助我们更好地了解我们的应用。

2.1.5 Axios

在我们使用html，css，js开发时，会使用原生的ajax来向后端请求数据，jQuery中可以用$.ajax()来请求数据，而Axios与上述两种功能类似，用来请求后台数据，Axios是一个基于promise的HTTP库，可以用在浏览器和node.js中，有以下几个特性：1.从浏览器中创建XMLHttpRequests，2.从node.js创建http请求，3.支持Promise API，4.拦截请求和响应，5.转换请求数据和响应数据，6.取消请求，7.自动转换JSON数据，8.开客户端支持和防御XSRF

2.1.6 Vue Router

路由是根据不同的url地址展示不同的内容或页面，前端路由就是把不同路由对应不同的内容或页面的任务交给前端来做，之前是通过服务器根据url的不同返回不同的页面实现的。我们在单页面应用，大部分页面结构不变，只改变部分内容时，使用前端路由。前端路由的优点有：用户体验好，不需要每次都从服务器全部获取，快速展现给用户；缺点有：不利于SEO，使用浏览器的前进、后退键的时候都会重新发送请求，没有合理地利用缓存，单页面无法记住之前滚动的位置，无法在前进，后退的时候记住滚动的位置；

Vue Router是Vue.js官方的路由管理器。它和Vue.js的核心深度集成，让构建单页面应用变得易如反掌。包含的功能有：1.嵌套的路由/视图表；2.模块化的、基于组件的路由配置；3.路由参数、查询、通配符；4.基于vue.js过度系统的视图过度效果；5.细粒度的导航控制；6.带有自动激活的CSS class的链接；7.HTML5历史模式或hash模式，在IE9中自动降级；8.自定义的滚动条行为。

2.1.7 ElementUI

随着前端技术的不断崛起，各种基于vue的组件库层出不穷，ElementUI是饿了么前端开源的一个基于vue的前端框架，已经帮我们封装好了一系列功能性的组件，比如栅格系统、表格、表单、树形菜单、通知、级联选择器、时间选择器、日期选择器、颜色选择器、警告、消息提示等。对于搞后台管理界面的项目，特别是不需要考虑兼容ie8、ie9以下的项目，ElementUI是一个不错的选择，而且ElementUI的文档写得十分详尽，参照demo可以很快上手。

2.1.8 Node.js

Node.js的诞生很大程度上归功于v8引擎的出现，我们都知道计算机处理器智能识别机器语言，而JavaScript是一门高级语言，计算机并不能直接读懂。所以我们需要所谓的引擎来将其转化成计算机所能理解的语言。v8引擎是由Google推出的，为其浏览器Chrome所设计的开源JavaScript引擎。得益于JIT，编译模式的改变与编译阶段的优化，JavaScript的性能得到了一个飞跃。除了对JavaScript性能的大幅提升，v8引擎也提供了“嵌入”的功能，使得开发者也可以在自己的c++程序中使用“嵌入”的v8引擎，从而高效地编译JavaScript，并加入c++的feature。于是，借助于v8种种便利的功能，Node.js诞生了。Node.js是一项服务器技术。我们都知道客户端提出服务请求，而服务器端负责处理请求并提供服务。而对于互联网来说，在Node.js之前JavaScript是一项完全的客户端技术，被用于浏览器中实现各种动画，对DOM的操作等等。而后端，即服务端则是由PHP、Python、Ruby、Java等等语言来实现。Node.js的出现，使得前后端使用同一种语言，统一模型的梦想得以实现。

2.1.9 Expreess

Express 是一个简单而灵活的Node.js Web应用程序开发框架，提供了一系列强大特性帮助你创建各种Web应用，和丰富的HTTP工具。使用Express可以快速地搭建一个完整功能的网站。Express框架的核心特性有：可以设置中间件来响应HTTP请求；定义了路由表用于执行不同的HTTP请求动作；可以通过向模板传递参数来动态渲染HTML页面。

2.1.10 MongoDB

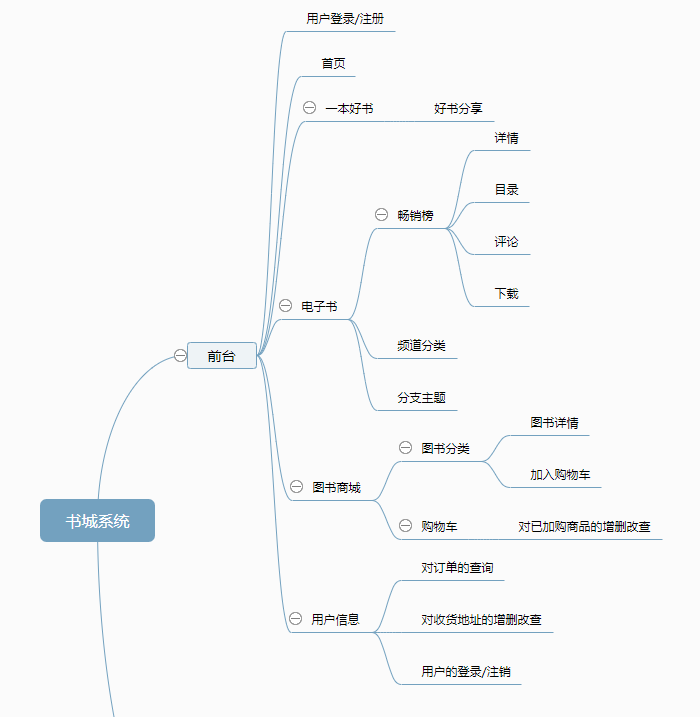
MongoDB是用C++语言编写的非关系型的数据库。关系型数据库中的表都是存储一些结构化的数据，每条记录的字段和组成都一样，即使不是每条记录都需要所有的字段，但数据库会为每条数据分配所有的字段。而非关系型数据库以键值对存储，它的结构不固定，每一条记录可以有不一样的键，每条记录可以根据需要增加一些自己的键值对，这样就不会局限于固定的结构，可以减少一些空间的开销。MongoDB的特点是高性能、易部署、易使用，存储数据十分方便，主要特性有：面向集合存储，易于存储对象类型的数据；模式自由；支持动态查询；支持完全索引，包含内部对象；支持复制和故障恢复；使用高效的二进制数据存储，包括大型对象；文件存储格式为BSON（一种JSON的扩展）。

2.1.11 Webpack

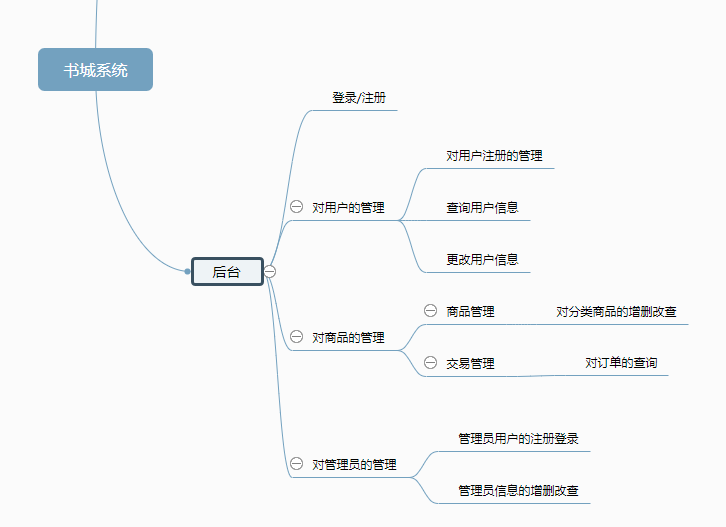
Webpack可以看做是模块打包机：它做的事情是，分析现有项目结构，找到JavaScript模块以及其它的一些浏览器不能直接运行的拓展语言（Scss，TypeScript等），并将其转换和打包为合适的格式供浏览器使用。Webpack的工作方式是：把项目当做一个整体，通过一个给定的主文件(如index.js)，Webpack将从这个文件开始找到该项目的所有依赖文件，使用loaders处理它们，最后打包为一个（或多个）浏览器可识别的JavaScript文件。

## 2.2 书城功能结构图

书城功能结构图如图2-1图a和图b所示：



图a 功能结构图前台系统



图b 功能结构图后台系统

图2-1 书城功能结构图

# 第3章 书城前端功能实现与展示

## 3.1 登录注册功能

一个系统必须需要的就是登陆和注册功能，用户进入一个网站，需要先进行注册来拥有自己的账号，输入用户信息和用户密码来获得账号，进而通过账号和密码登录网站。在进行注册时，前端对注册信息做了验证，前端验证账号和密码的输入格式，账号必须为手机号码，密码不能为纯数字，且在注册中添加了验证码的验证，为了防止机器操作对服务器的恶意攻击。登录后可绑定自己的邮箱，之后可通过手机号码或者邮箱来进行登录，如果忘记密码，可通过手机短信验证或者邮箱验证找回账号密码。页面展示如图3-1的图a和图b所示：

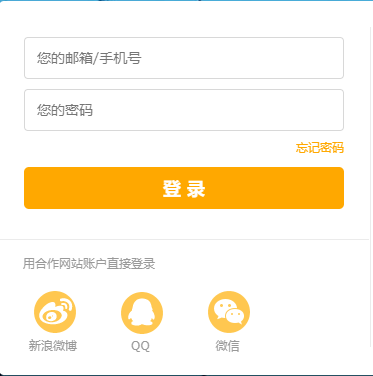
 

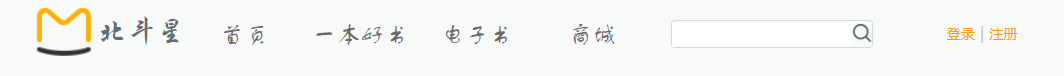
图3-1 注册和登录页面展示图

图a 注册页面

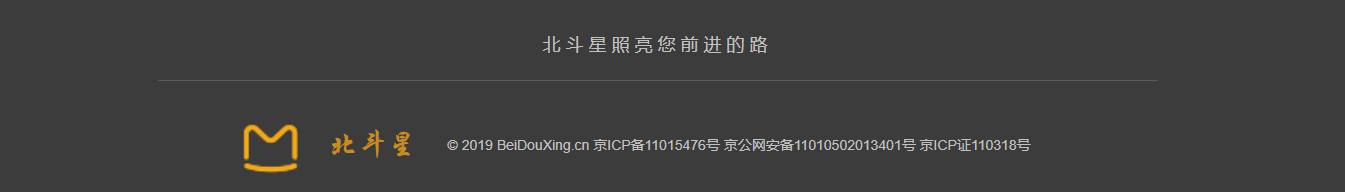
图b 登录页面

## 3.2 首页面功能

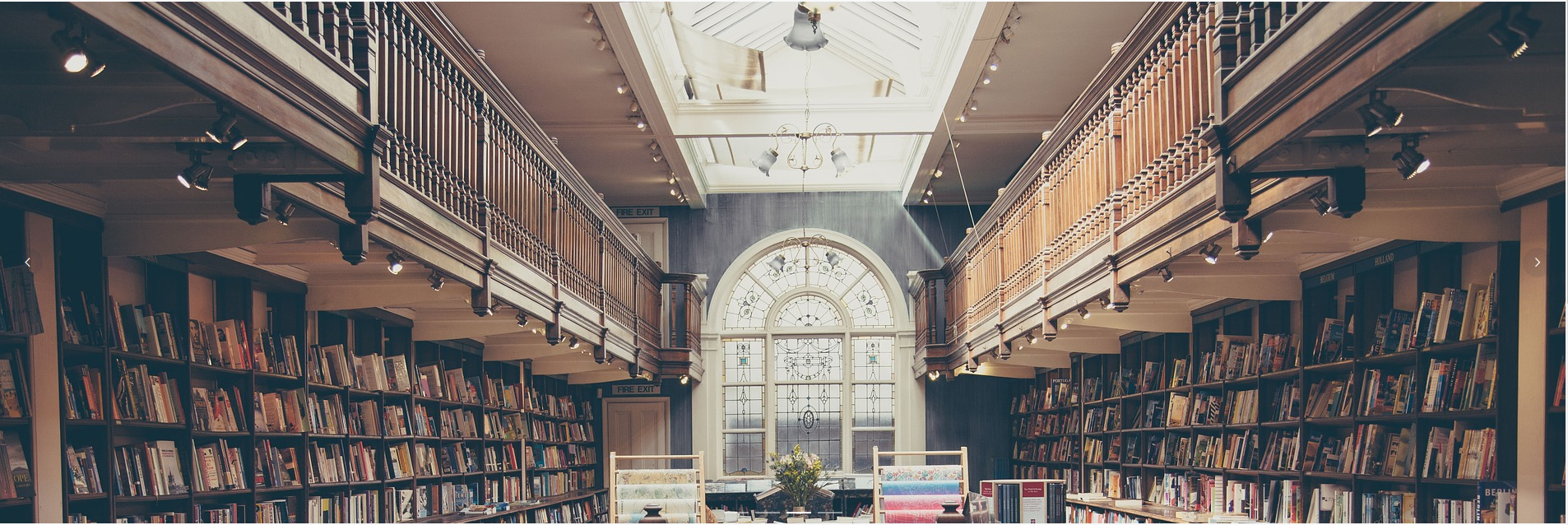
首页是一个网站的门面，首页应该给人舒服的感觉，首页中，分成了导航栏（如图a所示），内容和底部（如图b所示）。导航栏中有网站logo，首页，一本好书，电子书，和商城，以及搜索和登录注册，搜索用来输入关键字进行搜索想要找的书或者心得，登录和注册模块，当用户没有登录的时候，显示的是登录/注册，登录之后，显示用户信息，首页用来定位到首页面，一本好书用来定位到心得分享页面，电子书和商城也都分别定位到对应的页面。内容中，有一个和浏览器窗口相同宽度的轮播图（如图c所示），用来展示好看的书店图片，点击可进入到书店的介绍，人们可以看到自己喜欢的书店地址以及介绍。左边有作家专栏（如图d所示）和阅读心得（如图e所示），作家专栏用来放一些有名的作家，点击进去可以看到作家简介以及写过的书，阅读心得，用来放一些用户发表的心得笔记，点击可查看详细内容。右边是热门分享和最新发表（如图f所示），为用户发表的读书心得，当点击热门时，是点赞数最高的分享，点击最新分享时，是最近发表的分享。



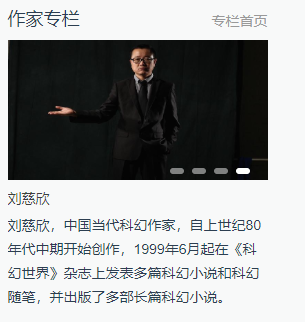
图a 导航栏



图b 底部



图c 轮播图模块



图d 作家专栏

图3-2 首页各模块展示图

## 3.3 好书分享功能

点击顶部导航栏的一本好书，进入到一本好书页面（如图a所示）来看到别的用户分享的好书文章，点击分享进入详情，可以查看整篇分享文章用来使用户了解别的写分享的作者推荐本书的原因，使自己更多地去了解是否喜欢该书。一本好书中，使用两种排序方式：最受欢迎的，和最新发表的，最受欢迎的，是指点赞数最高的分享，而最新发表的是距离当前时间最近的分享，也就是最新发表的分享。用户可以通过这两种排序方式，来选择看分享的方式，给用户提供了更大的方便。分享分为一个个小模块，每个小模块中包含图片、分享的文章名、内容摘抄、地点、昵称、看过的人数、以及点赞数。当然，用户可以在个人中心中发表自己的好书分享，这个在下一个模块介绍。



图a 一本好书页面

图3-3 分类和分类详情展示页面

## 3.4 电子书页面功能

点击导航栏电子书，会进入电子书页面，电子书页面中首先是排行榜(如图a所示)，有畅销榜，新书榜，和好评榜，排行榜是请求的当当的排行榜数据，图书排行榜是排行前10的书，当鼠标在书籍名上时，可以看到书籍名，作者，以及书籍封面，点击书籍名，进入书籍详情，有读书简介，目录和评论，而且用户也可以在底下点击发表评论来发表自己的评论，底下有电子书的链接，可以进行下载电子书离线查看。排行榜底下有一个查看详情，点击查看详情，便可以看到整个图书排行榜，前十五的书籍。

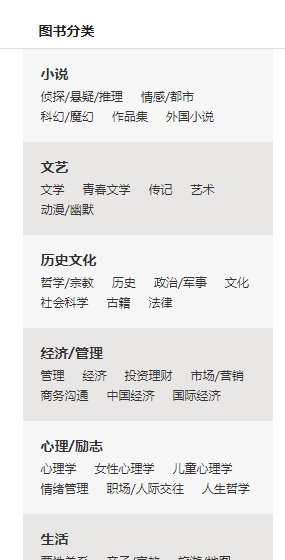


图3-4 电子书页面展示

图a 排行榜页面

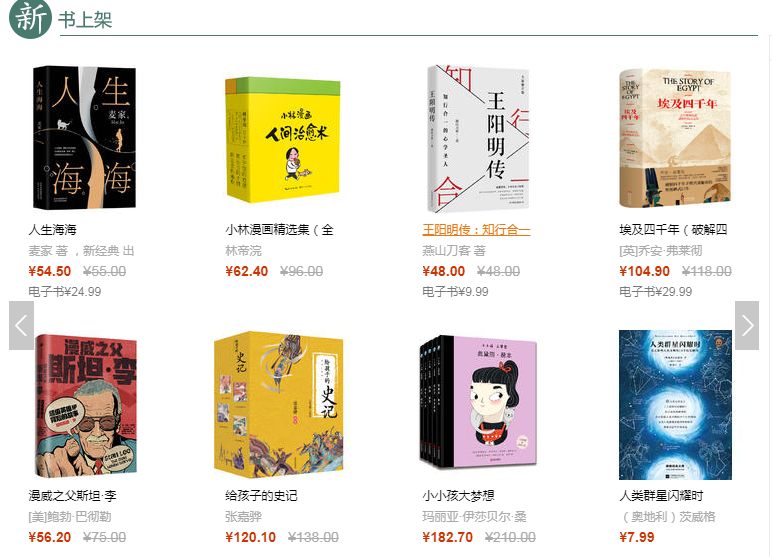
## 3.5 书城页面功能

点击导航栏书城，页面就会进入书城页面首页，书城是用来购买纸质书籍。书城首页中有图书分类列表(如图a所示)，可以根据分类来选择自己喜欢的类型的书籍，具体有教育、小说、文艺、青春文学、童书、人文社科、经营、成功/励志、生活、财经等类型，中间顶部有一个搜索，可以输入书籍名来搜索想要的书籍，或者输入作者，搜索到该作者出的书，底下是一个轮播图模块(如图b所示)，用来放热销书籍的广告，用户可以点击直接进入书籍介绍页面，轮播图下面是新书上架模块(如图c所示)，用来放最新的书籍列表，用户可以看到最新出来的书籍，点击进去购买。新书上架模块下面是独家特供模块，独家特供模块中的图书，是只有北斗星书城中有的，拿到独家出售版权的新书，右侧是新书预售和新书热卖榜，新书预售用来放还没有上架准备要上架的书籍，用户可以加入购物车等待新书开售进行购买，新书热卖榜中是销量最高的书籍排行，新书热卖榜分为总榜，童书，励志，传记和亲子五类，用户可以直接从热卖榜点击进去购买书籍。

图b 畅销书籍广告图

图a 图书分类列表图



图c 新书上架模块图

图3-5 书城页面功能展示图

## 3.6 购物系统功能分析

商城的主要作用是用来购物，那么购物的过程则是整个商城的重点，图书商城的购物和普通淘宝购物或者其他购物平台的购物流程是一样的，挑选物品，查看物品详情，最后确定购买后提交购买，支付，最后支付成功，获取订单，然后用户只需要等待发货就好。下面我以书城页面里面的新书上架中的《三体》来介绍购物的详细过程。如下图a所示的第一本书籍，名称是《三体》，假入这本书就是我确定要购买的，根据用户的需求，首先我需要先看一下物品的详细介绍，如图3-6中的图a所示这时只需要点击图片或者文字，页面会自动跳转到书籍详情页面，如图3-6中的图b所示，在书籍详情页上面对物品有一个简单的介绍和数量的选择，数量可以加可减，下拉页面会看到对商品的详细介绍，如图3-7中的图c和图d，图c是对物品进行了详细的介绍，通过介绍可以了解书籍的目录，分类，纸张，包装，书号，书籍详请页还有评论和问答，可以看到买家收到书籍后使用的感受，而图d则是问答，这样用户就可以提出自己对于书籍的疑虑邀请使用买过该书籍的人来作答，更大程度上使用户更好地了解书籍以及更准确地判断自己是否想要购买该书籍。详情页和对比页是通过选项卡来获取的，首先默认展示的是书籍详情，然后点击评论就可以进入评论页，点击问答进入问答页，点击书籍详情返回默认页，可以根据用户需求互相转换。关于书籍的介绍和所做对比数据都是后端存储在数据库里，前端通过需求调用数据库的字段值取相应的值展示在页面上，这一部分前后端的交互就显得十分重要，而且在前后端需要注意的是后端字段的确定，前端才能调用字段值。



图3-6 购物流程介绍图

图b 商品详情页

图a 各物品展示页

图a 书籍展示页

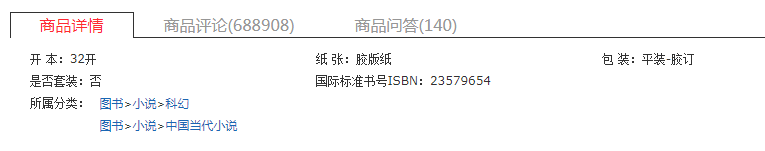
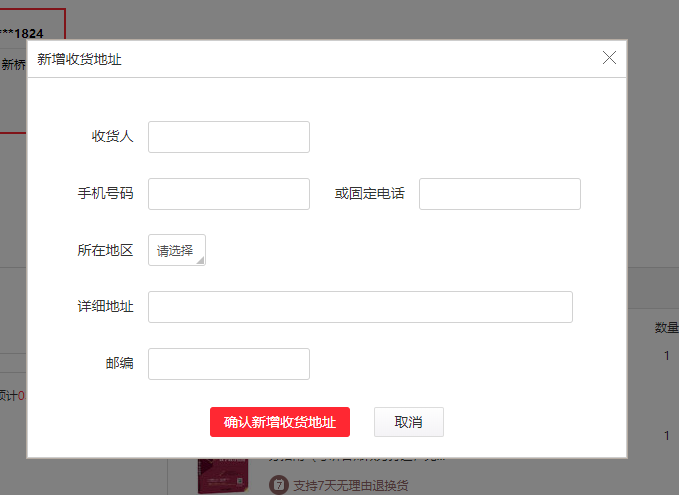
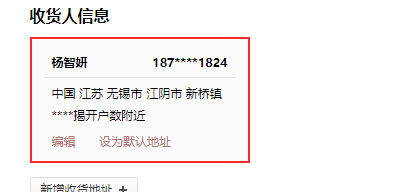


图3-6 购物系统功能

图b 详细介绍

如上分析，当用户了解完书籍的详情后，可以选择加入购物车选项然后进行结算或者直接购买，如图3-7中的图e则是物品加入购物车后的页面上展示，通过购物车选择结算最后进入订单页，一般收货地址是默认收货地址，如果此时没有地址，会让你填写地址，如图3-7中的图f，收货地址无误后点击确认提交，进行支付最后物品购买成功，成功后页面如图3-8中的图i。如果是在详情页直接下单，点击立即购买则会出现图3-7中的图f的页面，然后进行确认提交，提交后会出现支付页面，如图3-7中的图g和图h所示，当选择立即支付后会出现输入密码，输入密码完成则购买物品成功。如上所介绍的就是整个书城购书的整个流程。过程中数据都是通过数据库传输，前端使用Vue实现动态添加，前端调用后端使用的是Axios，此处需要注意的是详情页的获取数据，获取数据时需要先获得页面的参数，然后通过id参数是不同的，然后不同的id就可以把不同的数据获取到页面，然后当点击的id和后台id相等时，展示出相应物品的详情页。整个过程只限用在PC端，也是因为此次项目主要是基于PC端开发的书城系统。

图g 默认收货地址页

图f 填写收货地址页

图f 提交订单页

图3-7 购物流程介绍图

# 第4章 后端书籍管理功能与实现

## 4.1 后端书籍管理界面原理分析

呈现给客户的仅仅只是书城的前端页面及供用户使用的相关功能，但强大的商城如果没有后端管理系统，它就像一盘散沙，因此后端管理系统的作用在于开发让整个项目变得完善，并让项目能正常的去使用。本次商城后端商品管理系统是基于框架Element UI来实现页面及功能的，和其他框架一样，使用框架进行开发的好处在于效率高，少量代码实现复杂要求，Element UI是基于Vue框架用户界面插件的集合，它完整的组件集合让后台开发变得简单，而且页面设计也很美观，树形表、面板，下拉组合等的组件可以一起使用也可以分开单独去使用。

## 4.2 后端书籍管理界面分析

后端商品管理界面主要根据Element UI的Container布局容器，Container布局容器将页面分为Header,Aside,Main,Footer四个部分，Header为头部，可以用来作为导航栏或者页面头部，Aside为侧边栏，用来做侧边导航栏或者放置列表，Main为内容部分，Footer为底部。在本设计中，Header作为了头部，放置书城标题以及登录后的管理员信息，Aside作为侧边导航栏，Main中为切换展示的部分，Footer为底部，放置了版权信息，如图4-1所示示整个页面的整体布局，下面分区域介绍整个管理界面。

**4.2.1 页面头部分析**

页面的头部就是Container布局容器的Header区域，头部主要有北斗星书城后台管理系统标题，和右边对管理员用户登录信息的管理，将鼠标放在用户名上，可以看到关于该管理员的信息以及退出登录按钮。效果图如图4-2所示：

**4.2.2 页面侧边栏分析**

页面的侧边栏就是Container布局容器的Aside区域，页面的侧边导航栏通常放置在左侧，作为信息主干，也符合用户浏览习惯，侧边导航栏可以以更加干净优雅的方式呈现更多的链接，对于后台管理页面，使用侧边导航栏，可以很好地使页面简洁并美观。侧边栏中有书籍管理，和订单管理，书籍管理用来上架和下架书籍，以及书籍的发布和书籍信息的修改，订单管理中有订单管理和物流管理，订单管理中可以查看订单详情，比如支付方式，订单号，订单名称，时间等，也可以确认订单并输入物流单号来发货，物流管理中可以查看已发货但还没有确认收货的订单，历史订单中可以查看买家已经确认收货了的订单，如图4-3所示：

**4.2.3 页面内容分析**

页面的侧边栏就是Container布局容器的Main区域，页面内容用来展示在点击侧边栏后展示相对应的内容，当点击书籍管理下的书籍发布时，内容页面将展示书籍的发布页面，可以上传书籍照片并输入书籍名，目录，分类，纸张，包装，书号等书籍信息用于前台的展示。点击书籍管理时，展示书籍管理页，为书籍列表，列表上方有搜索框支持书籍名称和作者搜索，分页功能可选择以多少条来对表格进行分页，每一行列表都有对应的管理按钮，用来上架和下架商品，当商品发布成功之后，会进入书籍管理中，但是不会再前台展示，只有在书籍管理中，点击管理，然后上架该书籍，前台才可以看到改书籍的信息。页面内容中还有物流管理和订单管理页面，用来展示物流信息和订单信息。

**4.2.3 页面底部分析**

页面的侧边栏就是Container布局容器的Footer区域，底部是用来放关于系统的一些信息，比如服务器地址，备案号等。

# 第5章 项目疑难点和总结

随着互联网的发展以及人们对知识越来越多的渴求，网上书城也越来越多，通过此次的亲自设计书城管理系统，从最初的静态页面设计，到一步步的动态，连接后台，还是遇到很多很多的难题，不过经过一次次的努力，修改，百度，问题都是可以解决的。

## 5.1 书城前端界面系统问题分析及解决

**5.1.1 基本问题分析与解决**

（1）页面设计方面的问题：前端的设计对审美观还是有一定的要求，所以设计前端静态页面的时候在图片和色调的搭配上是有一定的难度的，不能过于单调，也不能太过于复杂，要考虑大众的眼光，页面呈现在用户眼前要考虑到不能让他们讨厌，要有吸引用户的特质。所以在页面设计这块是要经过一次次的修改，并多参考别人的网站设计，才能达到自己想要的效果。

（2）添加动态方面的问题：页面的动态功能很多，轮播图，页面跳转条件判断，前后台数据交互，最难的在于前后台数据交互，出现的问题常会有前台连接不上数据库，后台也接收不到前端传过去的数据，这个问题可能的原因就是Ajax这一部分请求没做合适，通常调试问题主要借助于浏览器的开发者模式，然后通过平台的console提示的错误进行错误排查并改正。

（3）网络响应速度问题：一个网站的加载速度和代码的编写也是有很大的关系，代码重复率比较高会导致响应速度降低，还有图片比较多也会影响响应速度，解决这些问题的方法在于尽可能的使用代码的共用，不要写多余的代码，图片在本地设置大小合适的尺寸，这样会减少图片太大造成的浪费带宽。

**5.1.2 设计中的难点分析及解决**

（1）设计中的难点主要在于前后端交互数据部分，前后端交互主要应用的技术是Axios，Axios进行数据的请求，加载，最后对数据进行解析，并展示在页面上。此处需要注意的是前端页面不仅仅是由简单的html和css静态完成的，而是通过Axios请求到的数据来驱动Vue创建节点，然后添加到页面，这里需要介绍的知识点就是关于Vue的数据驱动，传统的前端数据交互是Ajax从服务器端获取数据，然后操作DOM来改变视图；或者前端交互要改变数据时，又要再来一次上述步骤，而手动操作DOM是一个繁琐的过程且易出错。Vue.js是一个提供了MVVM风格的双向数据绑定的JavaScript库，专注于View层。它让开发者省去了操作DOM的过程，只需要改变数据，Vue会通过Dircetives指令，对DOM做一层封装，当数据发生改变会通知指令去修改对应的DOM，数据驱动DOM变化，DOM是数据的一种自然映射。Vue还会对操作进行监听，当视图发生改变时，vue监听到这些变化，从而改变数据，这样就行程了数据的双向绑定。当你把一个普通的 JavaScript 对象传给 Vue 实例的 data选项，Vue 将遍历此对象所有的属性，并使用 Object.defineProperty 把这些属性全部转为 getter/setter。Object.defineProperty 是 ES5 中一个无法 shim 的特性，这也就是为什么 Vue 不支持 IE8 以及更低版本浏览器的原因。用户看不到 getter/setter，但是在内部它们让 Vue 追踪依赖，在属性被访问和修改时通知变化。这里需要注意的问题是浏览器控制台在打印数据对象时 getter/setter 的格式化并不同，所以你可能需要安装 vue-devtools 来获取更加友好的检查接口。每个组件实例都有相应的 watcher 实例对象，它会在组件渲染的过程中把属性记录为依赖，之后当依赖项的setter被调用时，会通知watcher重新计算，从而致使它关联的组件得以更新。

（2）在电子书页面中，所遇到的难点就是在畅销榜中，我想要的效果在鼠标没有浮上去之前，显示第一个，当鼠标浮在哪一个书名上面时对应的图书图片展示，但我现在只实现了后者，没有默认显示的图书图片，目前问题正在进行解决中。

（3）书籍详情页的数据获取是一个难点，当在首页看到许许多多的书籍，但是每个书籍都有自己对应的详情页，但是如果每个书籍都写一个详情页不仅麻烦当数据不断的增加也不可能实现，所以此处需要的详情页其实只是一个页面，而内容则是随着点击物品的不同而从后台数据传过来的值不同。此处的难点就是后台传过来的数据是通过参数id的不同来传值的，我需要先获取到我点击详情时，我点击的书籍的id，然后将书籍id传给后台，后台返回对应id的数据，然后前台将数据展现出来。

（4）控制台经常出现的问题分析：

1

图 5-1 控制台错误样图1

如上图5-1所示，404（not found）代表所提示的内容没有被找到，控制台的好处就在于它会详细的指出错误代码在哪一部分的哪一行，而内容没被找到的原因就是该部分内容不存在或者路径出现问题。

1

图5-2 控制台错误样图2

如上图5-2所示，这个问题的意思就是调用的点击onclick事件为空，根据提示就可以去找该点击事件，而造成的原因就是触发该点击事件的事件元素未能成功找到，找到获取到的id，然后获取正确的事件，就可以成功解决该问题。

1

图5-3 控制台错误样图3

如上图5-3所示，这个问题和前面图13的问题是一样的，不同的是当页面滚动开始触发滚动事件后这个问题才会出现，而解决方法同上。

1. PC端页面需要注意事项：

本设计的书城主要是基于PC端的，但是不同的电脑屏幕大小像素还是有差距的，我们最首要的任务是保证网页在不同的屏幕尺寸下可以很好地展示，我是采用的固定px+百分比+最小宽高，给页面设置最小宽高，也就是最小的PC屏幕尺寸1280px，当显示窗口宽度小于1280px时，页面将显示滑动条来进行滑动显示。而大于时，使用百分比进行适配。

## 5.2 后端商品管理界面问题分析及解决

后台管理系统因为要用到json数据来完善树形，所以json数据的定义也是一个重点，但json数据并不是难点，只需要根据字节定义相应的值就好，值得注意的是json数据的值必须要用引号。而引用json数据完成树形的时候常出现树形结构并没有出现的现象，这种原因有可能是因为json数据定义的不合适或者加载json数据的路径不合适。如图5-4，出现404错误，就是因为json数据的路径不合适。

1

图5-4 后台管理系统控制台错误样图1

后台管理系统和后台数据库这一块紧密相连，有时候详情页的部分数据可能获取不到，遇到这种问题不能仅仅只从前端代码解决问题，如果仅仅咬住前端页面可能会走进死胡同，因为这种问题有可能是因为前后端的id不一致等各种因素造成数据加载失败。

## 5.3 总结

整个项目目前完成了三个系统模块，1.实现用户读书心得交流，2.实现电子书排行查看和电子书下载，3.图书商城中加入购物车和付款。其他模块（图书评论和电子书在线阅读）框架已基本完成，内容还有待继续完善，但是心得交流和电子书排行以及书城系统功能基本完善，还有待后期的发现问题并不断完善，通过本次整个书城系统的制作，对商城模块化的编程有了更深的理解，对所学知识点更是有了完整的巩固。也明白书到用时方恨少，和实践的重要性。

# 参考文献

[1] Mark Pilgrim．《[HTML5揭秘](http://www.amazon.cn/mn/detailApp/ref=as_li_qf_sp_asin_tl?_encoding=UTF8&tag=vastwork-23&linkCode=as2&asin=B004FJUVNI&camp=536&creative=3200&creativeASIN=B004FJUVNI)》[M].北京：电子工业出版社，2010-12．

[2] 陶国荣．《HTML5实战》[M].北京：电子工业出版社，2015-01-04．

[3] 吴丰．《[网页设计与制作](http://www.so.com/s?q=%E7%BD%91%E9%A1%B5%E8%AE%BE%E8%AE%A1%E4%B8%8E%E5%88%B6%E4%BD%9C&ie=utf-8&src=internal_wenda_recommend_textn)》[M].北京：清华大学出版社，2015-01．

[4] 《Vue.js权威指南》电子工业出版社，2016-08-01

[5] Nicholas C.Zakas著 刘振涛译 《深入理解ES6》 电子工业出版社，2017-07-01

[6] TGCode．《JavaScript半知半解》[M].北京：清华大学出版社，2017-01．

[7] [jim\_shen](https://home.cnblogs.com/u/shenxiaoquan/)．[淘淘商城项目：项目的Maven工程搭建](http://www.cnblogs.com/shenxiaoquan/p/6221082.html)[OL].http://www.cnblogs.

com/shenxiaoquan/p/6221082.html，2018-03-10．

[8] 囧囧． [原生JS实现触控滑动（swipe）图片轮播](https://segmentfault.com/a/1190000003110236)[OL].http://segmentfault.com/a/1190000003110236，2015-08-18

[9]Jeremy Keith 著 杨涛 王剑桥译《css权威指南（第3版）》中国电力出版社， 2010-4-15.

[10]张晓菲 单东林 《锋利的JQuery（第2版）》人民邮电出版社，2012-7-1.

[11] Chris Mills著.林雪玉 韦剑宾译 《CSS3实战：开发与实战》

致 谢

大学到现在马上就结束了，突然觉得时间过得好快，隐隐约约还记得大一时自己的兴奋，到大二的迷茫，大二第二学期进入实验室后自己对未来的规划，到大三对自己的工作有了一定的定位，大四顺利地找到自己心仪的工作，顺利完成论文，一切都得归功于老师和学长学姐的指导以及实验室浓郁的学习氛围，如果没有进实验室，我可能也没法那么明确地确定自己以后想干什么以及自己喜欢什么，当时在赵老师的帮助下，我进入网络实验室进行前端的学习，发现这就是自己所喜欢的事，想一直干下去。

大学的生活很美好，很自由，可以做自己想做的事，只要有想法想干，但是也正因为太自由，没有人指导容易陷入迷茫，不过只要你自己有方向后，你会发现，所有人都愿意帮你。当我确定做前端方向后，经常跟赵老师和学长学姐了解一些怎么学习，以及前端这条路上有什么坑我需要及时避免，他们给了我很大的帮助，使我在前端这条路上走得很顺畅。

而本次做的设计也是两年多的学习后开始的一项前端项目，基于web前端最流行的框架Vue开发而成，通过本次“北斗星书城”的开发，把之前所学的好多知识都进行了巩固，而过程中也感谢赵玉祥老师耐心的指导，不厌其烦的指出问题所在，也感谢在我遇到问题时给我思路的朋友，有句话说“一个人可以走的很快，但是一群人才能走得更远”，所以感谢团队的合作，让本次设计最后成功的展示给大家。